AD-A031 195 GEORGE WASHINGTON UNIV WASHINGTON D C PROGRAM IN LOG--ETC F/G 15/5
SENSITIVITY ANALYSIS FOR PARAMETRIC NON-LINEAR PROGRAMMING USIN--ETC(U)
JUL 76 R L ARMACOST, A V FIACCO N00014-75-C-0729

UNCLASSIFIED SERIAL-T-340 NL

1 OF 1
AD
A031195

END
DATE
FILMED
11 — 76

12

THE
GEORGE
WASHINGTON
UNIVERSITY

STUDENTS FACULTY STUDY R
ESEARCH DEVELOPMENT FUT
URE CAREER CREATIVITY CO
MMUNITY LEADERSHIP TECH
NOLOGY FRONTI ESIGN
ENGINEERING APP ENC
GEORGE WASHIN NIV

INSTITUTE FOR MANAGEMENT
SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING
AND APPLIED SCIENCE

(12)

SENSITIVITY ANALYSIS FOR PARAMETRIC NONLINEAR PROGRAMMING USING PENALTY METHODS

by

Robert L. Armacost
Anthony V. Fiacco

Serial T-340
30 July 1976

The George Washington University
School of Engineering and Applied Science
Institute for Management Science and Engineering

Program in Logistics
Contract N00014-75-C-0729
Project NR 347 020
Office of Naval Research

NONE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) Serial — T-340	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) SENSITIVITY ANALYSIS FOR PARAMETRIC NON-LINEAR PROGRAMMING USING PENALTY METHODS.		5. TYPE OF REPORT & PERIOD COVERED (9) SCIENTIFIC rept.
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) ROBERT L. ARMACOST, ANTHONY V. FIACCO		8. CONTRACT OR GRANT NUMBER(s) (15) N00014-75-C-0729
9. PERFORMING ORGANIZATION NAME AND ADDRESS THE GEORGE WASHINGTON UNIVERSITY PROGRAM IN LOGISTICS WASHINGTON, D. C. 20037		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (16) NR-347-020
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE (11) 30 July 1976
		13. NUMBER OF PAGES 40
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) OFFICE OF NAVAL RESEARCH CODE 430 D ARLINGTON, VA. 22217 (12) 44p.		15. SECURITY CLASS. (of this report) NONE
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

DISTRIBUTION OF THIS REPORT IS UNLIMITED.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

SENSITIVITY ANALYSIS	COMPUTER CODES
PARAMETRIC	INVENTORY MODEL
NONLINEAR PROGRAMMING	SUMT
PENALTY METHODS	KUHN-TUCKER TRIPLE

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Recently, it has been shown that a class of penalty function algorithms can readily be adapted to generate sensitivity analysis information for a large class of parametric nonlinear programming problems. In particular, estimates of the partial derivatives (with respect to the problem parameters) of the components of a solution vector and the optimal value function have been successfully calculated for a number of nontrivial examples. The approach has been implemented using the well known Sequential Uncon- NEXT Page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601

NONE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 337 LB

20. Abstract (Continued)

strained Minimization Technique (SUMT) computer program. This paper, a continuation and amplification of a recent paper by Armacost, gives a detailed summary of the significant underlying theoretical results, reviews recent additions to the computer program that include Lagrange multiplier sensitivity calculations, and elaborates on the kind of information that can be generated by further analyzing and interpreting results obtained in applying the technique to a well known inventory model.

TABLE OF CONTENTS

THE GEORGE WASHINGTON UNIVERSITY
School of Engineering and Applied Science
Institute for Management Science and Engineering

Program in Logistics

Abstract
of
Serial T-340
30 July 1976

SENSITIVITY ANALYSIS FOR PARAMETRIC NONLINEAR PROGRAMMING USING PENALTY METHODS

by

Robert L. Armacost*
Anthony V. Fiacco

Recently, it has been shown that a class of penalty function algorithms can readily be adapted to generate sensitivity analysis information for a large class of parametric nonlinear programming problems. In particular, estimates of the partial derivatives (with respect to the problem parameters) of the components of a solution vector and the optimal value function have been successfully calculated for a number of nontrivial examples. The approach has been implemented using the well known Sequential Unconstrained Minimization Technique (SUMT) computer program. This paper, a continuation and amplification of a recent paper by Armacost, gives a detailed summary of the significant underlying theoretical results, reviews recent additions to the computer program that include Lagrange multiplier sensitivity calculations, and elaborates on the kind of information that can be generated by further analyzing and interpreting results obtained in applying the technique to a well known inventory model.

---

*U.S. Coast Guard Headquarters, Washington, D.C. 20590.

THE GEORGE WASHINGTON UNIVERSITY
School of Engineering and Applied Science
Institute for Management Science and Engineering

Program in Logistics

SENSITIVITY ANALYSIS FOR PARAMETRIC NONLINEAR PROGRAMMING USING PENALTY METHODS

by

Robert L. Armacost
Anthony V. Fiacco

1. Introduction

Initial numerical results resulting from the implementation of a penalty function technique for obtaining sensitivity information in parametric nonlinear programming were given by Armacost and Fiacco (1974). The work is based on the theory developed by Fiacco and McCormick (1968) and extended by Fiacco (1973). This paper reports on refinements and extensions of the computational procedures implemented by Armacost and Mylander (1973) and Armacost (1976), using the SUMT-Version 4 computer code with the logarithmic-quadratic loss penalty function to estimate the partial derivatives of the solution point and the objective function optimal value, the derivatives here being taken with respect to the specified problem parameters.

Fiacco (1973) developed the necessary general formulas for the partial derivatives of the "optimal value function," the components of a local solution point and its associated optimal Lagrange multipliers, for a large class of parametric nonlinear programming problems composed of twice differentiable functions. He also obtained approximation formulas

in terms of the well known logarithmic-quadratic penalty function. Recently, Armacost and Fiacco (1975) particularized and simplified these formulas for various problem structures and developed formulas for the first and second derivatives of the optimal value function of the given problem. Additionally, Armacost and Fiacco (1976) have applied the general theory to easily prove the well known result that, when the parameters are the right-hand side components of the constraints, the optimal Lagrange multipliers give the gradient of the optimal value function (with respect to the parameters). Further, it was shown that the first derivatives of the Lagrange multipliers give the components of the Hessian of the optimal value function, and explicit formulas were developed for the Hessian in terms of the problem functions.

In their first report on computational experience, Armacost and Fiacco (1974) concentrated primarily on presenting computational experience associated with the calculation of the first derivatives of a local solution point. The practical implementability of the approach was demonstrated.

In a subsequent paper, Armacost (1976) reported on additional computational experience, focusing on the calculation of the derivatives of the optimal value function and the Lagrange multipliers, also implementing a potentially valuable refinement that allows for computerized screening for "key" parameters. This paper may be regarded as a continuation and amplification of the Armacost paper.

For problems involving a large number of parameters, a very large number of partial derivatives may be calculated if one proceeds indiscriminately. This is not only time-consuming, but may also be quite burdensome to a user who must evaluate the overall significance of the results. One

measure of the latter is the effect of a perturbation on the solution value. It is quite possible and often observed in practice that the optimal objective function value is much more sensitive to a few of the many parameters present. With this in mind, the method developed by Armacost and Fiacco (1975) to estimate the first order sensitivity of the optimal value function was incorporated in the computer program to provide an option for preliminary screening of the parameters to eliminate further calculations involving perturbations of parameters having "little" effect on the optimal value function. (A user can easily introduce his own criteria of significance in this determination.) Using the formulas developed by Fiacco (1973), a second option is included which permits the calculation of the sensitivity estimates for the Lagrange multipliers. The computer code and options used to accomplish these and other calculations are discussed in Section 3.

In Section 4, a sensitivity analysis is conducted for a multi-item inventory model developed by Schrady and Choe (1971) for the U.S. Navy. The example analyzed is the same small one treated by Schrady and Choe, though readily extended to a large-scale model. The results illustrate the potential value of a detailed automated post-optimality sensitivity analysis in practical situations, and hopefully dramatize the numerous rich interpretations and insights that can be derived from this information, as well as indicating the caution that must be taken in making valid inferences.

The recently obtained basic theoretical results validating the computational algorithm are summarized rather completely in the next section so that the paper might be self contained.

2. Supporting Theory

The parametric mathematical programming problems considered here are of the form

$$\begin{aligned} &\underset{x \,\varepsilon\, E^n}{\text{minimize}} \quad f(x,\varepsilon) \\ &\text{subject to} \quad g_i(x,\varepsilon) \geq 0 \ , \quad i=1,\ldots,m \ , \\ &\qquad\qquad\quad h_j(x,\varepsilon) = 0 \ , \quad j=1,\ldots,p \ , \end{aligned} \qquad P(\varepsilon)$$

where x is the usual vector of variables and ε is a k-component vector of numbers called "parameters." It is desired ultimately to develop a complete characterization of a solution $x(\varepsilon)$ of Problem $P(\varepsilon)$ as a function of ε. In our current work, we have concentrated on certain recently computationally tractable measures of change in a solution as ε is perturbed from a specified value. (Without loss of generality, we assume that the specified value is $\varepsilon=0$.)

When certain assumptions are satisfied, Fiacco (1973) and Armacost and Fiacco (1975) have characterized the "first order sensitivity" of a "Kuhn-Tucker Triple" and the first and second order sensitivity of the optimal value function of Problem $P(\varepsilon)$. (These quantities are defined as the theory is presented.) Additionally, they have developed formulas for efficiently estimating this sensitivity when the logarithmic-quadratic loss penalty function algorithm is used to solve Problem $P(\varepsilon)$. The main theoretical results are summarized here.

The Lagrangian for Problem $P(\varepsilon)$ is defined as

$$L(x,u,w,\varepsilon) \equiv f(x,\varepsilon) - \sum_{i=1}^{m} u_i g_i(x,\varepsilon) + \sum_{j=1}^{p} w_j h_j(x,\varepsilon) \ ,$$

where u_i, $i=1,\ldots,m$ and w_j, $j=1,\ldots,p$ are "Lagrange multipliers" associated with the inequality and equality constraints, respectively. Any vector $(\bar{x},\bar{u},\bar{w})$ satisfying the usual (first order) Kuhn-Tucker conditions (Fiacco and McCormick, 1968) of Problem $P(\bar{\varepsilon})$ is called a Kuhn-Tucker triple.

The following four assumptions are sufficient to establish the results and are assumed to hold throughout the paper:

A1 -- The functions defining Problem $P(\varepsilon)$ are twice continuously differentiable in (x,ε) in a neighborhood of $(x^*,0)$.

A2 -- The second order sufficient conditions for a local minimum of Problem $P(0)$ hold at x^* with associated Lagrange multipliers u^* and w^*.

A3 -- The gradients $\nabla_x g_i(x^*,0)$ (i.e., $(\partial g_i(x^*,0)/\partial x_1,\ldots,\partial g_i(x^*,0)/\partial x_n)^T$, the superscript T denoting transposition) for all i such that $g_i(x^*,0) = 0$, and $\nabla_x h_j(x^*,0)$, $j=1,\ldots,p$ are linearly independent.

A4 -- Strict complementary slackness holds at $(x^*,0)$ (i.e., $u_i^* > 0$ for all i such that $g_i(x^*,0) = 0$).

Theorem 1: (Local characterization of a Kuhn-Tucker triple (Fiacco, 1973) of Problem $P(\varepsilon)$.) If assumptions A1, A2, A3 and A4 hold for Problem $P(\varepsilon)$ at $(x^*,0)$, then

(a) x^* is a local isolated minimizing point of Problem $P(0)$ and the associated Lagrange multipliers u^* and w^* are unique;

(b) for ε in a neighborhood of 0, there exists a unique, once continuously differentiable vector function $y(\varepsilon) =$

$(x(\varepsilon),u(\varepsilon),w(\varepsilon))^T$ satisfying the second order sufficient conditions for a local minimum of Problem $P(\varepsilon)$ such that $y(0) = (x^*,u^*,w^*)^T = y^*$ and hence, $x(\varepsilon)$ is a locally unique, local minimum of Problem $P(\varepsilon)$ with associated unique Lagrange multipliers $u(\varepsilon)$ and $w(\varepsilon)$; and

(c) for ε near 0 , the set of binding inequalities is unchanged, strict complementary slackness holds for $u_i(\varepsilon)$ for i such that $g_i(x(\varepsilon),\varepsilon) = 0$, and the binding constraint gradients are linearly independent at $x(\varepsilon)$.

This result provides a characterization of a local solution of Problem $P(\varepsilon)$ and its associated optimal Lagrange multipliers near $\varepsilon=0$. It generalizes a theorem first presented by Fiacco and McCormick (1968, Theorem 6) and is closely related to a generalization of the same theorem provided independently by Robinson (1974). It shows that the Kuhn-Tucker triple $y(\varepsilon)$ is unique and well behaved, under the given conditions. Since $y(\varepsilon)$ is once differentiable, the partial derivatives of the components of $y(\varepsilon)$ are well defined. This fact and Assumption A1 also mean that the functions defining Problem $P(\varepsilon)$ are once continuously differentiable functions of ε along the "solution trajectory" $x(\varepsilon)$ near $\varepsilon=0$, and the Lagrangian is a once continuously differentiable function of ε along the "Kuhn-Tucker point trajectory."

We are thus motivated to determine a means to calculate the various partial derivatives, since this yields a first order estimate of the locally optimal Kuhn-Tucker triple and the problem functions near $\varepsilon=0$.

Denote by $\nabla_\varepsilon x(\varepsilon) \equiv (\partial x_i(\varepsilon)/\partial\varepsilon_j)$, $i=1,\ldots,n$, $j=1,\ldots,k$, the $n \times k$ matrix of partial derivatives of $x(\varepsilon)$ with respect to ε , and define $\nabla_\varepsilon u(\varepsilon)$ and $\nabla_\varepsilon w(\varepsilon)$ in a similar fashion. We then define

$\nabla_\varepsilon y(\varepsilon) \equiv (\nabla_\varepsilon^T x(\varepsilon), \nabla_\varepsilon^T u(\varepsilon), \nabla_\varepsilon^T w(\varepsilon))^T$, an $(n+m+p) \times k$ matrix.

When $y(\varepsilon)$ is available, $\nabla_\varepsilon y(\varepsilon)$ can be calculated by noting that Conclusion (b) of the theorem implies the satisfaction of the Kuhn-Tucker conditions for $P(\varepsilon)$ at $y(\varepsilon)$ near $\varepsilon=0$, i.e.,

$$\begin{aligned} \nabla_x L[x(\varepsilon),u(\varepsilon),w(\varepsilon),\varepsilon] &= 0 , \\ u_i(\varepsilon)g_i[x(\varepsilon),\varepsilon] &= 0 , \quad i=1,\ldots,m , \\ h_j[x(\varepsilon),\varepsilon] &= 0 , \quad j=1,\ldots,p . \end{aligned} \tag{1}$$

Since the Jacobian $M(\varepsilon)$ of this system with respect to (x,u,w) (i.e., the matrix obtained by differentiating the left side of (1) with respect to the components of (x,u,w)) is nonsingular under the given assumptions, the total derivative of the system with respect to ε is well defined and must equal zero. This yields

$$M(\varepsilon)\nabla_\varepsilon y(\varepsilon) = N(\varepsilon) ,$$

where $N(\varepsilon)$ is the negative of the Jacobian of the Kuhn-Tucker system with respect to ε , and hence

$$\nabla_\varepsilon y(\varepsilon) = M(\varepsilon)^{-1} N(\varepsilon) .$$

The class of algorithms based on twice continuously differentiable penalty functions can be used <u>without additional assumptions and without requiring $y(\varepsilon)$</u> to provide an estimate of $\nabla_\varepsilon y(\varepsilon)$. Furthermore, most of the information required to make the estimate <u>is already available</u> in the typical implementations of these algorithms. Here, we use the logarithmic-quadratic penalty function for Problem $P(\varepsilon)$ (Fiacco and McCormick, 1968) defined as

$$W(x,\varepsilon,r) \equiv f(x,\varepsilon) - r \sum_{i=1}^{m} \ln g_i(x,\varepsilon) + (1/2r) \sum_{j=1}^{p} h_j^2(x,\varepsilon) . \quad (2)$$

Under the given assumptions, the following facts are known for Problem P(0) from penalty function theory (Fiacco and McCormick, 1968, Theorems 10 and 17):

(1) For $r > 0$ and small, there exists a unique once continuously differentiable vector function $x(0,r)$ such that $x(0,r)$ is a locally unique minimizing point of $W(x,0,r)$ in $R^{\#}(0) \equiv \{x: g_i(x,0) > 0 , i=1,\ldots,m , \text{ and } h_j(x,0) = 0 , j=1,\ldots,p\}$ and such that $x(0,r) \to x(0,0) = x^*$;

(2) $\lim_{r\to 0} r \sum_{i=1}^{m} \ln g_i[x(0,r)] = 0$;

(3) $\lim_{r\to 0} (1/2r) \sum_{j=1}^{p} h_j^2[x(0,r),0] = 0$; and

(4) $\lim_{r\to 0} W[x(0,r),0,r] = f(x^*,0)$.

The following theorem extends these results for Problem $P(\varepsilon)$, where ε is allowed to vary in a neighborhood of 0 , and provides a basis for approximating the sensitivity information associated with Problem $P(\varepsilon)$. The notation $\nabla_x^2 W$ denotes the matrix of second partial derivatives of W with respect to x .

<u>Theorem 2</u>: (Relationship of solutions of Problem $P(\varepsilon)$ and minima of $W(x,\varepsilon,r)$, (Fiacco 1973).) If Assumptions A1 - A4 hold, then in a neighborhood about $(\varepsilon,r) = (0,0)$ there exists a unique once continuously differentiable vector function $y(\varepsilon,r) = [x(\varepsilon,r),u(\varepsilon,r)w(\varepsilon,r)]^T$ satisfying

$$\nabla_x L(x,u,w,\varepsilon) = 0 \quad ,$$
$$u_i g_i(x,\varepsilon) = r \quad , \quad i=1,\ldots,m \; ,$$
$$h_j(x,\varepsilon) = w_j r \; , \quad j=1,\ldots,p \; ,$$

with $y(0,0) = (x^*,u^*,w^*)$ and such that, for any (ε,r) near $(0,0)$ and $r > 0$, $x(\varepsilon,r)$ is a locally unique unconstrained local minimizing point of $W(x,\varepsilon,r)$, $g_i[x(\varepsilon,r),\varepsilon] > 0$, $i=1,\ldots,m$, and $\nabla_x^2 W[x(\varepsilon,r),\varepsilon,r]$ is positive definite.

Corollary 2.1: (Convergence of estimates using $W(x,\varepsilon,r)$, (Fiacco, 1973).) If Assumptions A1, A2, A3 and A4 hold for Problem $P(\varepsilon)$, then for any ε near 0 ,

(a) $\lim_{r\to 0^+} y(\varepsilon,r) = y(\varepsilon,0) = y(\varepsilon)$, the Kuhn-Tucker triple characterized in Theorem 1; and

(b) $\lim_{r\to 0^+} \nabla_\varepsilon y(\varepsilon,r) = \nabla_\varepsilon y(\varepsilon,0) = \nabla_\varepsilon y(\varepsilon)$.

This result motivates use of $\nabla_\varepsilon y(\varepsilon,r)$ to estimate $\nabla_\varepsilon y(\varepsilon)$, when ε is near 0 and r is near 0 , once $y(\varepsilon,r)$ is available. Theorem 2 provides the basis for an efficient calculation of $\nabla_\varepsilon y(\varepsilon,r)$. Since, at a local solution point $x(\varepsilon,r)$ of $W(x,\varepsilon,r)$, it follows that

$$\nabla_x W[x(\varepsilon,r),\varepsilon,r] = 0 \; , \tag{3}$$

we can differentiate (3) with respect to ε to obtain

$$\nabla_x^2 W[x(\varepsilon,r),\varepsilon,r]\nabla_\varepsilon x(\varepsilon,r) + \nabla_\varepsilon(\nabla_x W[x(\varepsilon,r),\varepsilon,r]) = 0 \; . \tag{4}$$

By Theorem 2, $\nabla_x^2 W$ is positive definite for (ε,r) near $(0,0)$ and $r > 0$, so $\nabla_x^2 W$ has an inverse and $\nabla_\varepsilon x(\varepsilon,r) = -\nabla_x^2 W[x(\varepsilon,r),\varepsilon,r]^{-1} \cdot \nabla_{\varepsilon x}^2 W[x(\varepsilon,r),\varepsilon,r]$.

Also, since

$$u_i(\varepsilon,r) = r/g_i(x(\varepsilon,r),\varepsilon) \ , \quad i=1,\ldots,m \ , \tag{5}$$

and

$$w_j(\varepsilon,r) = h_j(x(\varepsilon,r),\varepsilon)/r \ , \quad j=1,\ldots,p \ , \tag{6}$$

for (ε,r) near $(0,0)$ and $r > 0$, these equations can be differentiated with respect to ε to obtain

$$\nabla_\varepsilon u_i(\varepsilon,r) = -(r/g_i^2)[\nabla_x g_i(x(\varepsilon,r),\varepsilon) \cdot \nabla_\varepsilon x(\varepsilon,r) + \partial g_i(x(\varepsilon,r),\varepsilon)/\partial\varepsilon] \ , \tag{7}$$

$$\nabla_\varepsilon w_j(\varepsilon,r) = (1/r)[\nabla_x h_j(x(\varepsilon,r),\varepsilon) \cdot \nabla_\varepsilon x(\varepsilon,r) + \partial h_j(x(\varepsilon,r),\varepsilon)/\partial\varepsilon] \ . \tag{8}$$

Solving (4) and calculating (7) and (8) then yields the components of $\nabla_\varepsilon y(\varepsilon,r)$, which can be used to estimate $\nabla_\varepsilon y(\varepsilon)$ for (ε,r) near $(0,0)$.

The next results extend this theory to an analysis of the optimal value function of Problem $P(\varepsilon)$ along the Kuhn-Tucker point trajectory $[x(\varepsilon),u(\varepsilon),w(\varepsilon)]^T$.

The optimal value function is defined as:

$$f^*(\varepsilon) \equiv f[x(\varepsilon),\varepsilon] \ , \tag{9}$$

and the "optimal value Lagrangian" is defined as:

$$L^*(\varepsilon) = L[x(\varepsilon),u(\varepsilon),w(\varepsilon),\varepsilon] \ . \tag{10}$$

<u>Theorem 3</u>: (First and second order changes in the optimal value function, Armacost and Fiacco (1975).) If assumptions A1 - A4 hold for Problem $P(\varepsilon)$, then for ε near 0 , $f^*(\varepsilon)$ is a twice continuously differentiable function of ε , and

(a) $f^*(\varepsilon) = L^*(\varepsilon)$;

(b) $$\nabla_\varepsilon f^*(\varepsilon) = \nabla_\varepsilon L(x,u,w,\varepsilon) \Big|_{(x,u,w)=(x(\varepsilon),u(\varepsilon),w(\varepsilon))}$$

$$= \nabla_\varepsilon f(x,\varepsilon) - \sum_{i=1}^{m} u_i \nabla_\varepsilon g_i(x,\varepsilon)$$

$$+ \sum_{j=1}^{p} w_j \nabla_\varepsilon h_j(x,\varepsilon) \Big|_{(x,u,w)=(x(\varepsilon),u(\varepsilon),w(\varepsilon))} ;$$

(c) $\nabla_\varepsilon^2 f^*(\varepsilon) = \nabla_\varepsilon(\nabla_\varepsilon L(x(\varepsilon),u(\varepsilon),w(\varepsilon),\varepsilon)^T)$.

The logarithmic-quadratic loss penalty function (2) can also be used to provide estimates of the first and second order sensitivity of the optimal value function. Let the optimal value penalty function be defined as $W^*(\varepsilon,r) \equiv W(x(\varepsilon,r),\varepsilon,r)$.

Theorem 4: (First and second order sensitivity of $W^*(\varepsilon,r)$ and estimates for $f^*(\varepsilon)$, Armacost and Fiacco (1975).) If Assumptions A1 - A4 hold for Problem $P(\varepsilon)$, then for (ε,r) near $(0,0)$ and $r > 0$, $W^*(\varepsilon,r)$ is a twice continuously differentiable function of ε and

(a) $\lim_{r\to 0^+} W^*(\varepsilon,r) = L^*(\varepsilon) = f^*(\varepsilon)$;

(b) $$\nabla_\varepsilon W^*(\varepsilon,r) = \nabla_\varepsilon L(x,u,w,\varepsilon) \Big|_{(x,u,w)=(x(\varepsilon,r),u(\varepsilon,r),w(\varepsilon,r))} ;$$

(c) $$\lim_{r\to 0^+} \nabla_\varepsilon W^*(\varepsilon,r) = \nabla_\varepsilon L(x(\varepsilon),u(\varepsilon),w(\varepsilon),\varepsilon) = \nabla_\varepsilon f^*(\varepsilon) ; \tag{11}$$

(d) $\nabla_\varepsilon^2 W^*(\varepsilon,r) = \nabla_\varepsilon(\nabla_\varepsilon L(x(\varepsilon,r),u(\varepsilon,r),w(\varepsilon,r),\varepsilon)^T)$;

(e) $\lim_{r\to 0^+} \nabla_\varepsilon^2 W^*(\varepsilon,r) = \nabla_\varepsilon^2 f^*(\varepsilon)$.

This result provides a justification for estimating $f^*(\varepsilon)$, $\nabla_\varepsilon f^*(\varepsilon)$ and $\nabla_\varepsilon^2 f^*(\varepsilon)$ by $W^*(\varepsilon,r)$, $\nabla_\varepsilon W^*(\varepsilon,r)$ and $\nabla_\varepsilon^2 W^*(\varepsilon,r)$, respectively, when r is positive and small enough.

Since Corollary 2.1 and continuity imply that $\lim_{r \to 0^+} f(x(\varepsilon,r),\varepsilon) = f^*(\varepsilon)$, another estimate of the optimal value function (9) is provided by $f^{\#}(\varepsilon,r) \equiv f(x(\varepsilon,r),\varepsilon)$ when $r > 0$ and small. Direct application of the chain rule for differentiation then yields, for $x = x(\varepsilon,r)$,

$$\nabla_\varepsilon f^{\#}(\varepsilon,r) = \nabla_x f(x,\varepsilon)\nabla_\varepsilon x(\varepsilon,r) + \nabla_\varepsilon f(x,\varepsilon) . \tag{12}$$

Under the given assumptions, continuity also assures that $\nabla_\varepsilon f^{\#}(\varepsilon,r) \to \nabla_\varepsilon f^*(\varepsilon)$ as $r \to 0^+$. Thus, both $\nabla_\varepsilon f^{\#}(\varepsilon,r)$ and $\nabla_\varepsilon W^*(\varepsilon,r)$ are estimates of $\nabla_\varepsilon f^*(\varepsilon)$ for r sufficiently small.

It should be noted that these estimates are functionally related since

$$\nabla_\varepsilon W^*(\varepsilon,r) = \nabla_\varepsilon f^{\#}(\varepsilon,r) - \sum_{i=1}^{m} u_i(\nabla_x g_i \nabla_\varepsilon x(\varepsilon,r) + \nabla_\varepsilon g_i) + \sum_{j=1}^{p} w_j(\nabla_x h_j \nabla_\varepsilon x(\varepsilon,r) + \nabla_\varepsilon h_j) \Big|_{x=x(\varepsilon,r)} .$$

From this expression, it is clear that $\nabla_\varepsilon f^{\#}(\varepsilon,r)$ is the better estimate of $\nabla_\varepsilon f^*(\varepsilon)$, the remaining terms in $\nabla_\varepsilon W^*(\varepsilon,r)$ simply constituting "noise" that is eliminated as $r \to 0^+$. However, by using the expression for $\nabla_\varepsilon W^*(\varepsilon,r)$ given by (11), $\nabla_\varepsilon W^*(\varepsilon,r)$ can be evaluated <u>without necessitating</u> the calculation of $\nabla_\varepsilon x(\varepsilon,r)$, which is required to compute (12). Thus, the cruder but computationally much cheaper estimate of $\nabla_\varepsilon f^*(\varepsilon)$ given by Equation (11) has now been introduced as an option in the computer program as a preliminary screening device to identify crucial parameters. Restriction of subsequent calculations to these parameters, and other calculations such as the sharper estimate of $f^*(\varepsilon)$ given by (12) are provided as additional options.

In summary, the basis for the estimation procedure utilized here for a specific problem, say Problem P(0), is the minimization of the penalty function $W(x,\varepsilon,r)$ given by (2). This yields a point $x(\varepsilon,r)$ which may be viewed as an estimate of a (local) solution x^* of Problem P(0). The associated optimal Lagrange multipliers u^* and w^* are then estimated by using the relationships given in (5) and (6), respectively. The first partial derivatives of these quantities with respect to ε are then obtained by first solving (4) and then applying (7) and (8). The estimate $f(x(\varepsilon,r),\varepsilon)$ of $f^*(\varepsilon)$ is immediately available when $x(\varepsilon,r)$ has been determined, and the two estimates of $\nabla_\varepsilon f^*(\varepsilon)$ given by (11) and (12) were already discussed. Various options and computer codes implementing the procedure are discussed in the next section.

3. User Options, Computer Codes and an Example

The basic SUMT-Version 4 computer program and instructions for its use are described in Mylander, Holmes and McCormick (1971). The basic sensitivity analysis subroutines, user instructions, and instructions for integrating the sensitivity package with the SUMT-Version 4 code are described in Armacost and Mylander (1973).

Briefly, the conduct of a sensitivity analysis is controlled by the variable NEXOP3 which is given a value on the "Second Option Card" in the SUMT input data deck. There are four choices: no sensitivity analysis, a sensitivity analysis at the final subproblem, a sensitivity analysis at each subproblem along the penalty function minimizing trajectory, or a sensitivity analysis at the final subproblem for a range of differencing increments. In conjunction with this option, two additional options are added here and come into play whenever a sensitivity analysis is conducted.

The first option (technically Option 4) is controlled by the variable NEXOP4 and determines whether the partial derivatives of the Lagrange multipliers will be calculated. When the calculation is done, the formulas developed by Fiacco (1973) are used.

The second option added here (Option 5) permits a screening of the parameters to reduce the number of partial derivatives which are estimated by limiting further analysis to parameter changes which affect the optimal value of the objective function by an amount exceeding 0.1 percent of its current value. This option is controlled by the variable NEXOP5. The estimate of sensitivity of the optimal value function with respect to a particular parameter under this option is calculated using the Armacost and Fiacco (1975) result which involves the partial derivative of the Lagrangian taken with respect to the parameter under consideration.

Subroutines LMULT and PRESEN and related coding in Subroutine SENS implement Option 4 and Option 5, respectively. Subroutines SENS, LMULT and PRESEN are listed in Appendix A. Specific instructions for using these two options in conjunction with the "Second Option Card" are given below in Table 1. This information should be added to Table 5 in Mylander, Holmes and McCormick (1971).

As an illustration of the kind of information that can be generated, consider the following simple (convex) parametric nonlinear programming problem,

$$\begin{aligned} &\text{minimize} \quad f(x,\varepsilon) = x_1 + \varepsilon_2 x_2 \\ &\text{subject to} \quad g_1(x,\varepsilon) = \varepsilon_1^2 - x_1^2 - x_2^2 \geq 0 \,. \end{aligned} \qquad A(\varepsilon)$$

for $\varepsilon_1 > 0$ and ε_2 not restricted.

TABLE 1

THE SECOND OPTION CARD

Option	Column	Value	Meaning
4	28	=0	Do not estimate the partial derivatives of the estimates of the Lagrange multipliers.
		=1	Estimate the partial derivatives of the estimates of the Lagrange multipliers whenever a sensitivity analysis of the solution point is conducted.
5	35	=0	Estimate the partial derivatives of the optimal value function and eliminate those parameters which do not affect the optimal value function from subsequent sensitivity calculations.
		=1	Estimate the partial derivatives of the optimal value function with respect to all parameters, but continue all subsequent sensitivity calculations with respect to all parameters.
		=2	Do not estimate the partial derivatives of the optimal value function first. Conduct the sensitivity analysis with respect to all parameters.

To dramatize the complexity of the relationships that can arise even in such trivial problems, we first give the solution in closed form. Application of the Kuhn-Tucker conditions yields the following:

$$x(\varepsilon) = \begin{bmatrix} x_1(\varepsilon) \\ x_2(\varepsilon) \end{bmatrix} = \begin{bmatrix} -\dfrac{\varepsilon_1}{\sqrt{1+\varepsilon_2^2}} \\[2ex] -\dfrac{\varepsilon_1\varepsilon_2}{\sqrt{1+\varepsilon_2^2}} \end{bmatrix},$$

$$u(\varepsilon) = \sqrt{1 + \varepsilon_2^2} \,/\, 2\varepsilon_1 \ ,$$

and

$$g_1(x(\varepsilon),\varepsilon) = 0 \ .$$

(It may be readily verified that Assumptions A1 - A4 are satisfied.)

From the expression for $x(\varepsilon)$, we find that

$$f^*(\varepsilon) = -\varepsilon_1\sqrt{1 + \varepsilon_2^2} \ .$$

Taking partial derivatives with respect to ε_1 and ε_2 , we obtain

$$\nabla_\varepsilon x(\varepsilon) = \frac{1}{\sqrt{1 + \varepsilon_2^2}} \begin{bmatrix} -1 & \dfrac{\varepsilon_1\varepsilon_2}{(1 + \varepsilon_2^2)} \\ -\varepsilon_2 & \dfrac{-\varepsilon_1}{(1 + \varepsilon_2^2)} \end{bmatrix} ,$$

$$\nabla_\varepsilon f^*(\varepsilon) = (\partial f^*(\varepsilon)/\partial\varepsilon_1,\ \partial f^*(\varepsilon)/\partial\varepsilon_2)$$

$$= \left(-\sqrt{1 + \varepsilon_2^2}\ ,\ -\varepsilon_1\varepsilon_2/\sqrt{1 + \varepsilon_2^2}\right),$$

and

$$\nabla_\varepsilon u^*(\varepsilon) = \left(-\sqrt{1 + \varepsilon_2^2}\ /2\varepsilon_1^2\ ,\ \varepsilon_2/\left(2\varepsilon_1\sqrt{1 + \varepsilon_2^2}\right)\right).$$

Suppose we are particularly interested in the values $\varepsilon_1=2$, $\varepsilon_2=1$.

Evaluation of the above expressions yields

$$f^* = -2\sqrt{2}\ , \qquad \nabla_\varepsilon f^* = (-\sqrt{2}, -\sqrt{2})\ ,$$

$$x^* = \begin{bmatrix} -\sqrt{2} \\ -\sqrt{2} \end{bmatrix}, \qquad \nabla_\varepsilon x^* = \begin{bmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ -\sqrt{2}/2 & -\sqrt{2}/2 \end{bmatrix},$$

$$u^* = \sqrt{2}/4 \quad , \quad \nabla_{\varepsilon} u^* = (-\sqrt{2}/8, \ \sqrt{2}/8) \ .$$

The numerical results obtained by the computer program are included in Table 2 for the optimal value function and Lagrange multiplier sensitivity. The values of the first order optimal value function sensitivity computed both by the chain rule (Equation (12)) and by taking partial derivatives of the Lagrangian with respect to the parameters (Equation (11)) are presented for comparison. As indicated previously and expected, the chain rule estimates are sharper. The subproblems correspond to minimizations of the penalty function $W(x,\varepsilon,r)$ for several values of $r > 0$, the value of r decreasing as the subproblem index increases. The values of f are given by $f(x(\varepsilon,r),\varepsilon)$, while the values of u and $\nabla_{\varepsilon} u$ were obtained by applying formulas (5) and (7). The components of $x(\varepsilon)$ and $\nabla_{\varepsilon} x(\varepsilon)$ were calculated also but are not portrayed.

TABLE 2

SENSITIVITY ESTIMATES FOR PROBLEM $A(\varepsilon)$, WHERE $\varepsilon=(2,1)$

Subproblem	f	Lagrangian $\partial f/\partial\varepsilon_1$	Lagrangian $\partial f/\partial\varepsilon_2$	Chain Rule $\partial f/\partial\varepsilon_1$	Chain Rule $\partial f/\partial\varepsilon_2$	u	$\partial u/\partial\varepsilon_1$	$\partial u/\partial\varepsilon_2$
1	-1.9999	-1.9999	- .9999	-1.3333	-1.3333	.4999	-.3333	.1666
2	-2.5393	-1.5440	-1.2947	-1.4087	-1.4088	.3860	-.2100	.1761
3	-2.7765	-1.4439	-1.3833	-1.4139	-1.4139	.3609	-.1844	.1767
4	-2.8128	-1.4243	-1.4064	-1.4142	-1.4142	.3560	-.1790	.1768
5	-2.8245	-1.4127	-1.4123	-1.4142	-1.4142	.3531	-.1768	.1768
6	-2.8274	-1.4137	-1.4137	-1.4142	-1.4142	.3532	-.1767	.1768
7	-2.8282	-1.3899	-1.4141	-1.4142	-1.4142	.3475	-.1737	.1768
Analytical	-2.8282	-1.4142	-1.4142	-1.4142	-1.4142	.3537	-.1769	.1768

The results are typical of those that have been obtained to date, convergence of the sensitivity estimates being reliable and stable with less stability noted in the estimates of the optimal Lagrange multipliers. It appears that the sensitivity estimates will converge satisfactorily whenever the SUMT algorithm succeeds in determining a good estimate of a Kuhn-Tucker point.

4. A Large-scale Multi-item Inventory Model

Traditionally, inventory models have been formulated to minimize some function of the ordering, holding and shortage (or backorder) costs subject to various constraints. Schrady and Choe (1971) have formulated an inventory model which appears to have much greater relevance for an inventory system in a noncommercial environment, such as institutional or military. The costs used in the traditional models may be quite artificial and the real objective of the system is often maximization of a measure of readiness or service, here assumed to be equivalent to minimization of stockouts. In addition, the stock points of such supply systems are inevitably constrained by investment and reorder work load limitations.

Schrady and Choe's multi-item inventory system assumes these constraints along with the specific objective of minimizing the total time-weighted shortages. The decision variables are taken to be the "reorder quantities" and the "reorder points," respectively, how much to order and when to order each item in the inventory. A three-item example problem was solved by Schrady and Choe (1971) using the SUMT computer code (Mylander, et al., 1971). Subsequently, McCormick (1972) showed how the special structure of this inventory model can be used to facilitate the use of the SUMT code to solve very large inventory problems. He also

extended the model to include constraints on storage volume and the probability of depletion of critical items.

The model and example presented here are the original ones due to Schrady and Choe. The computer program described in the preceding section was used to solve the example and calculate the partial derivatives of various quantities of interest, with respect to each parameter involved in defining the model. (The analysis can be applied to the extended model without difficulty.)

Detailed development of the model is beyond the scope of this paper. The interested reader is referred to the Schrady-Choe and McCormick papers. Here, we give a summary treatment of the various conditions and relationships upon which the model is based. We then tabulate the results obtained in solving the resulting nonlinear programming problem and applying the sensitivity analysis methodology. A number of observations and interpretations are offered to illustrate the many uses to which the sensitivity information might be applied.

It is assumed that the amount of each item in inventory is always known, that all demand which occurs when the on-hand stock is zero is backordered, and that the demand which occurs during the time between the placement of an order and its receipt by the stock point (i.e., the "lead time demand") is normally distributed with known mean μ_i and variance σ_i^2.

For the ith item, let

c_i = item unit cost (in dollars),

λ_i = mean demand per unit time (in units),

r_i = reorder point,

Q_i = reorder quantity,

$\phi(x)$ = the Normal (0,1) density function,

$\Phi(z) = \int_z^\infty \phi(x)dx$ = the Normal (0,1) complementary cumulative distribution function.

In addition, Let K_1 be the investment limit in dollars, K_2 the number of orders per unit of time that constitutes reorder work load limit, and N the total number of items in the inventory.

It can be shown that the expected time-weighted shortage of item i at any point in time is given by

$$B_i(Q_i,r_i) = \frac{1}{Q_i}[\beta_i(r_i) - \beta_i(Q_i+r_i)]$$

where

$$\beta_i(r_i) = \frac{1}{2}\left[\sigma_i^2 + (r_i-\mu_i)^2\right]\Phi\left(\frac{r_i-\mu_i}{\sigma_i}\right) - \frac{\sigma_i}{2}(r_i-\mu_i)\phi\left(\frac{r_i-\mu_i}{\sigma_i}\right).$$

The expected on-hand inventory of item i is given by $r_i + Q_i/2 - \mu_i + B_i(Q_i,r_i)$ and the expected number of orders placed per unit time for item i is λ_i/Q_i .

Using the above expressions and assumptions, Schrady and Choe (1971) indicate that meaningful approximations of the given quantitites are obtained even when the second term is dropped from the expression for the expected shortages, and when the last term is dropped from the expression for expected on-hand inventory. The given assumptions and simplifications then lead readily to the following nonlinear programming problem (which Schrady and Choe (1971) proved convex),

$$\underset{Q,r}{\text{minimize}} \quad Z(Q,r) \equiv \sum_{i=1}^{N} \beta_i(r_i)/Q_i$$

$$\text{subject to} \quad g_1(Q,r) \equiv K_1 - \sum_{i=1}^{N} c_i(r_i + Q_i/2 - \mu_i) \geq 0 \ ,$$

$$g_2(Q,r) \equiv K_2 - \sum_{i=1}^{N} \lambda_i/Q_i \geq 0 \ , \qquad \text{(SC)}$$

with r_i unrestricted in sign, $Q_i \geq 0$, $i=1,\ldots,N$, $Q = (Q_1,\ldots,Q_N)^T$, $r = (r_1,\ldots,r_N)^T$, and g_1 and g_2 representing the investment and work load constraints, respectively.

The problem data for the Schrady-Choe three-item example and the initial starting point for the SUMT program are shown in Table 3. As indicated in the table, the lead time demands and standard deviations, the item unit costs and mean demands, and the investment and work load limits are all treated as parameters in conducting the sensitivity analysis.

Table 4 gives the computer solution and Table 5 the final estimate of the first partial derivatives of the optimal value function Z^* with respect to the problem parameters. Relative to the criterion used in the computer program, the Table 5 results indicate that the optimal value function is sensitive to parameters K_2 , c_1 , σ_2 , c_2 , σ_3 and c_3 . Many inferences are possible. For example, the fact that the solution is particularly sensitive to the values of the standard deviations of the lead time demand of items 2 and 3 might indicate that, since these parameters were obtained by sampling, additional sampling of these lead time demands may very well be warranted to reduce the associated standard deviations.

Table 5 also suggests that the optimal solution value is very sensitive to all of the item costs. If the structure of Problem (SC) is examined, this result may at first appear contradictory since the c_i appear only

TABLE 3

INVENTORY PROBLEM DATA

	QUANTITY	ITEM i: 1	2	3	
PARAMETERS	μ_i	100	200	300	(MEAN OF LEAD-TIME DEMAND)
	σ_i	100	100	200	(S.D. OF LEAD-TIME DEMAND)
	c_i	1	10	20	(ITEM UNIT COST - DOLLARS)
	λ_i	1,000	1,500	2,000	(MEAN DEMAND/UNIT TIME)
	K_1	$ 8,000			(INVESTMENT LIMIT)
	K_2	15 re-orders/unit time			(RE-ORDER WORKLOAD LIMIT)
VAR.	Q_i^0	600	270	300	(AMOUNT ORDERED)
	r_i^0	200	250	400	(RE-ORDER LEVEL)

TABLE 4

SOLUTION AND LAGRANGE MULTIPLIERS

	QUANTITY	ITEM i		
		1	2	3
VAR.	Q_i^*	533	246	285
	r_i^*	253	277	437
L.M.	u_1^*	.0052		
	u_2^*	.6230		

TABLE 5

OPTIMAL VALUE FUNCTION DERIVATIVES

PARTIALS	ITEM i		
	1	2	3
$\partial Z^*/\partial \mu_i$	-.0000	-.0003	-.0008
$\partial Z^*/\partial \sigma_i$	.0119	.0897[a]	.1729[a]
$\partial Z^*/\partial c_i$	2.1713[a]	1.0345[a]	1.4452[a]
$\partial Z^*/\partial \lambda_i$	.0012	.0025	.0022
$\partial Z^*/\partial K_1$		-.0052	
$\partial Z^*/\partial K_2$		-.6230[a]	

[a]Deemed "significant" by criterion, $|\Delta_1 Z^*|/Z^* > .001$ for a unit change in the given parameter, where $\Delta_1 Z^*$ is the estimated first order change in Z^* . This criterion was selected arbitrarily for illustrative purposes. Criteria appropriate to the particular application can be selected by a user.

in the investment constraint and the optimal value function, according to Table 5, is apparently not very sensitive to the investment limit K_1. The problem is one of precise interpretation. The partial derivatives measure rate of change. But inspection of the investment constraint $g_1(Q,r)$ at (Q^*,r^*) reveals that the change in an item cost c_i by any amount Δc_i has the same effect on the constraint as a change in the investment limit K_1 of $-(r_i^* + Q_i^*/2 - \mu_i)\Delta c_i$. Since the quantity in parentheses may be verified from Tables 3 and 4 to be much greater than one for all i, it follows that the effect of changing any c_i by any increment δ will be much greater on the constraint (and hence, on the optimal value Z^*, since the constraint is binding) than the effect of changing K_1 by the same amount δ. This implies that $|\partial Z^*/\partial c_i| >$ $|\partial Z^*/\partial K_1|$ for each i and, in fact, it can be shown here that $\partial Z^*/\partial c_i =$ $-(r_i^* + Q_i^*/2 - \mu_i)\partial Z^*/\partial K_1$, so that the relationships indicated are indeed precisely verified.

The above observations might also suggest that some care must be taken in interpreting the results. Changes in the parameter associated with the largest (in absolute value) partial derivative will give the greatest local change in the optimal value of the objective function, compared to a change of the same magnitude in any other parameter taken individually. This follows because either the objective function and/or some of the constraints (as above) are most significantly affected by this parameter change at the current solution. General rules have not been given for selection of optimal changes in the parameters, i.e., for determining the optimal magnitude and combination of such changes. It is well beyond the scope of this paper to pursue this "macro-analysis" determination, though it should be noted that the greatest local rate of decrease in the

optimal value function is along the direction of the negative of the gradient of this function in parameter space (i.e., along the vector composed of the negative of the components of the partial derivatives with respect to the various parameters). A user would nonetheless have to determine the feasibility of this direction of change and, if feasible, the optimal move along this vector, taking into account other factors such as the relative "cost-effectiveness" of any schedule of changes in any model parameter.

Referring back to Table 4, we note that the Lagrange multiplier u_2^* is much greater than u_1^*. Recalling the "sensitivity" interpretation of Lagrange multipliers, which holds under the present conditions, it follows that $u_1^* = -\partial Z^*/\partial K_1$ and $u_2^* = -\partial Z^*/\partial K_2$. This conclusion is consistent with the result obtained in Table 5, and it means that the work load constraint g_2 is by far the more effective in determining the minimum number of expected time-weighted shortages at the current value of the parameters, e.g., a <u>small</u> increase in K_2 will have a greater effect on reducing Z^* than a small increase in K_1. Nonetheless, a user must again simultaneously consider the comparative costs involved in making <u>finite</u> changes, in conjunction with their expected effects, to arrive at an optimal marginal improvement based on this first order information. The sensitivity information is valuable, but requires some care in exploiting.

Table 6 gives the estimates of the first derivatives of the optimal reorder quantities Q_i and reorder points r_i with respect to each of the problem parameters. This is extremely detailed information which gives an indication of how the components of the solution vector itself will change as the various parameters change. In particular, this information

TABLE 6

SOLUTION POINT SENSITIVITY

PARTIALS	ITEM i		
	1	2	3
$\partial Q_i/\partial K_2$	- 47.3187	- 18.7610	- 14.9065
$\partial r_i/\partial K_2$	5.2265	6.1961	9.9671
$\partial Q_i/\partial c_1$	-208.8688	15.3140	14.3755
$\partial r_i/\partial c_1$	- 31.7918	- 10.3425	- 20.0020
$\partial Q_i/\partial \sigma_2$	- .8469	.2271	- .1084
$\partial r_i/\partial \sigma_2$	- .1273	1.0337	- .4919
$\partial Q_i/\partial c_2$	8.1908	- 4.9719	3.8522
$\partial r_i/\partial \sigma_2$	- 2.6783	- 7.2676	- 7.1087
$\partial Q_i/\partial \sigma_3$	- 1.2374	- .4033	.5843
$\partial r_i/\partial \sigma_3$	- .2611	- .3839	.0446
$\partial Q_i/\partial c_3$	- .1670	.4442	- .4251
$\partial r_i/\partial c_3$	- 2.3072	- 3.1702	- 12.1523

can be used to obtain a first order estimate of the solution vector of a problem involving different parameter values, having obtained a solution for a given set of parameters.

The partial derivatives of the Lagrange multipliers with respect to the parameters are given in Table 7. Again, these can be used to obtain first order estimates of the Lagrange multipliers of a problem with different parameter values. In particular, the relative effects of the constraints on the optimal value of problems involving different parameter values can be estimated. Furthermore, it can be shown that the partial derivatives of the multipliers with respect to K_1 and K_2 yield the <u>second</u> partial derivatives of the optimal value function with respect to the parameters K_1 and K_2, under the present conditions. Thus, the kind of information given in Table 7 can be used to provide a <u>second order</u> estimate of the optimal value function Z^* for different values of these parameters.

To illustrate and test the application of the type of information provided here, the first partial derivatives with respect to c_1 of the optimal value function Z^*, the solution components Q_i and r_i, and the Lagrange multipliers u_1 and u_2, were used to give a first order (Taylor's Series) estimate of the corresponding solution values associated with the problem where the given value of c_1 was increased by one dollar. These estimates were compared with the respective values of the solution obtained by actually solving the perturbed problem. The results are summarized in Table 8. Though the perturbation is large (the parameter being increased by 100% of its current value), the estimates are seen to be extremely accurate with the exception of the estimated reorder quantity Q_1. Many uses could be made of the estimated solution; e.g., should it be desirable to solve the perturbed problem accurately, it would be

TABLE 7

L.M. SENSITIVITY

OPTIMAL L.M. DERIVATIVES		
PARTIALS	CONSTRAINT i	
	1: INVESTMENT	2: WORKLOAD
$\partial u_i^*/\partial K_2$	-.0002	-.1382
$\partial u_i^*/\partial c_1$	.0006	.1635
$\partial u_i^*/\partial \sigma_2$	.0000	.0006
$\partial u_i^*/\partial c_2$	.0002	.0489
$\partial u_i^*/\partial \sigma_3$	.0000	.0020
$\partial u_i^*/\partial c_3$	.0003	.0323

computationally extremely advantageous to use the estimated solution as a starting point.

The complete exploitation of this sensitivity analysis information now available will depend largely on user interest and ingenuity.

TABLE 8

FIRST ORDER ESTIMATES FOR A UNIT INCREASE OF PARAMETER c_1

QUANTITY $F(\varepsilon^1)$	ESTIMATE $F(\varepsilon^1)_1$ †	ACTUAL	% ABS. ERROR
$Z^*(\varepsilon^1)$	15.159	14.996	1.08
$Q_1(\varepsilon^1)$	324	412	21.36
$r_1(\varepsilon^1)$	221	229	3.49
$Q_2(\varepsilon^1)$	261	257	1.56
$r_2(\varepsilon^1)$	267	268	.37
$Q_3(\varepsilon^1)$	299	297	.67
$r_3(\varepsilon^1)$	417	420	.71
$u_1(\varepsilon^1)$	.0058	.0057	1.75
$u_2(\varepsilon^1)$	.7865	.7671	1.94

† $F(\varepsilon^1)_1 = F(\varepsilon^0) + (\varepsilon^1-\varepsilon^0)^T \nabla_\varepsilon F(\varepsilon^0)$

$= F(\varepsilon^0) + (1)\partial F(\varepsilon^0)/\partial c_1$.

REFERENCES

ARMACOST, ROBERT L. (1976). Computational experience with optimal value function and Lagrange multiplier sensitivity in NLP. Technical Paper Serial T-335. Program in Logistics, The George Washington University, Washington, D.C.

ARMACOST, ROBERT L. and ANTHONY V. FIACCO (1974). Computational experience in sensitivity analysis for nonlinear programming. Mathematical Programming 6 301-326.

ARMACOST, ROBERT L. and ANTHONY V. FIACCO (1975). Second-order parametric sensitivity analysis in NLP and estimates by penalty function methods. Technical Paper Serial T-324. Institute for Management Science and Engineering, The George Washington University, Washington, D.C.

ARMACOST, ROBERT L. and ANTHONY V. FIACCO (1976). NLP sensitivity for R.H.S. perturbations: A brief survey and recent second-order extensions. Technical Paper Serial T-334. Institute for Management Science and Engineering, The George Washington University, Washington, D.C.

ARMACOST, ROBERT L. and W. CHARLES MYLANDER (1973). A guide to a SUMT-Version 4 computer subroutine for implementing sensitivity analysis in nonlinear programming. Technical Paper Serial T-287. Program in Logistics, The George Washington University, Washington, D.C.

COLVILLE, A. R. (1968). A comparative study of nonlinear programming codes. IBM New York Scientific Center Technical Report 320-2947.

FIACCO, ANTHONY V. (1973). Sensitivity analysis for nonlinear programming using penalty methods. Technical Paper Serial T-275. Institute for Management Science and Engineering, The George Washington University, Washington, D.C.

FIACCO, ANTHONY V. and GARTH P. McCORMICK (1968). Nonlinear Programming: Sequential Unconstrained Minimization Techniques. John Wiley and Sons, Inc., New York.

McCORMICK, GARTH P. (1972). Computational aspects of nonlinear programming solutions to large scale inventory problems. Technical Memorandum Serial TM-63488. Program in Logistics, The George Washington University, Washington, D.C.

MYLANDER, W. CHARLES, RAYMOND L. HOLMES and GARTH P. McCORMICK (1971). A guide to SUMT-Version 4: The computer program implementing the sequential unconstrained minimization technique for nonlinear programming. RAC-P-63, Research Analysis Corporation, McLean, Virginia.

ROBINSON, S. M. (1974). Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear programming algorithms. Mathematical Programming **7** 1-16.

SCHRADY, D. A. and U. C. CHOE (1971). Models for multi-item continuous review inventory policies subject to constraints. Naval Res. Logist. Quart. **18** 541-563.

van de PANNE, C. and W. POPP (1963). Minimum-cost cattle feed under probabilistic protein constraints. Management Sci. **9** 405-430.

APPENDIX A

SUBROUTINES SENS, LMULT AND PRESEN

```
      SUBROUTINE SENS
C
C                     15 MARCH 1972
C
C  THIS VERSION OF THE SENSITIVITY ANALYSIS SUBROUTINE IS USED TO
C  COMPUTE THE DIRECTIONAL DERIVATIVES OF X AND F WITH RESPECT TO
C  CERTAIN PARAMETERS CODED IN THE ARRAY PAR(20).  THE DIRECTIONAL
C  DERIVATIVES ARE ESTIMATED FOR ONE PARAMETER AT A TIME WITH NPAR BEING
C  THE NUMBER OF PARAMETERS INVOLVED IN THE SENSITIVITY ANALYSIS.  THE
C  USE OF THE PARAMETERS PAR(20) MUST BE CONSISTENT THROUGHOUT THE
C  USER'S SUBROUTINES.
C  THE SUBROUTINE IS USED FOR A SENSITIVITY ANALYSIS AT THE FINAL SUB-
C  PROBLEM OR FOR A SENSITIVITY ANALYSIS AT EACH SUBPROBLEM ALONG THE
C  MINIMIZING TRAJECTORY.  DPAR(20) IS THE ARRAY OF DIFFERENCING
C  INTERVALS CORRESPONDING TO THE PARAMETERS PAR(20).  DPAR(20) IS
C  ASSIGNED VALUES IN SUBROUTINE PARDIF.
C  THIS APPROACH TO SENSITIVITY ANALYSIS IS DUE TO A. V. FIACCO. THE
C  FIRST VERSION WAS CODED BY R. CAUSEY.  THE SECOND VERSION WAS CODED
C  BY W. C. MYLANDER.  THIS IS THE THIRD VERSION WHICH IS AN EXTENSION
C  OF THE SECOND VERSION TO PERMIT SENSITIVITY ANALYSIS ALONG THE
C  MINIMIZING TRAJECTORY, AND WAS CODED BY R. L. ARMACOST.
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 RHOIN,RATIO,EPSI,THETAO,XEP1,XEP2
      COMMON/SHARE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/EQAL/H,H1,MZ
      COMMON/OPTNS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
      COMMON/VALUE/F,G,RC,RSIGMA,FJ(40),RHO
      COMMON/REST/DELX(20),DELXO(20),RHOIN,RATIO,EPSI,THETAO,
     1NTCTR,NUMINI,X1(20),X2(20),X3(20),XR2(20),XR1(20),PR1,
     2PR2,P1,F1,RJ1(40),DOTT,PGRAD(20),DIAG(20),
     3PREV3,ADELX,PSIG1,G1,NPHASE,NSATIS
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
      COMMON/EXPOPT/NEXOP1,NEXOP2,NEXOP3,NEXOP4,NEXOP5,XEP1,XEP2
      DIMENSION DELT(20),DELTX(20),X2H(20),X3H(20)
      DIMENSION DELMU(40),DU(40),KTEST(20)
      NEXOP4 = 1
      NEXOP5 = 0
      DO 5 I=1,N
      DELTX(I) = DELX(I)
    5 DELT(I) = DELXO(I)
      CALL STORE
      AVAL = 1.0E-10
      MPMZ = M + MZ
      ISENS = ISENS + 1
      CALL PARDIF
C  WRITE OUT X, RHO AND PAR.
      WRITE(6,10) RHO
   10 FORMAT(1H //30X,20HSENSITIVITY ANALYSIS     //
     15X,22HTHE VALUE OF R(RHO) IS ,E12.5)
      WRITE(6,20)
   20 FORMAT(/5X, 64HTHE POINT AT WHICH THE ESTIMATE OF SENSITIVITY WILL
     1BE MADE IS      )
      DO 40 I=1,N,6
      II=MIN0(I+5,N)
      WRITE(6,30) ((J,X(J)), J=I,II)
   30 FORMAT( 6(2X,2HX(,I2,2H)=,G14.7) )
   40 CONTINUE
      WRITE(6,50) ((I,PAR(I),DPAR(I)), I=1,NPAR)
```

```
      50 FORMAT(45H)      PARAMETER VALUE   DIFFERENCING INTERVAL /
     1 (I4,2X,G14.5,6X,G14.5) )
         WRITE(6,55)
      55 FORMAT(//)
C  EVALUATE F, G AND H.
         CALL RESINT(0,F)
         IF(MPMZ.EQ.0) GO TO 85
         DO 80 J=1,MPMZ
         CALL RESINT(J,RJ(J))
      80 CONTINUE
         IF(NEXOPS.EQ.3) GO TO 85
         CALL PRESEN(DU,KTEST)
C  COMPUTE DEL F.
      85 CALL GRAD1(0)
         DO 90 I=1,N
         X3H(I) = DEL(I)
      90 CONTINUE
C  COMPUTE (DEL)**2 P - STORED IN A.
         CALL SECORD(2)
C  PERFORM THE L-U DECOMPOSITION OF A.
         DO 100 I=1,N
         DELX(I)=0.0
     100 CONTINUE
         NP=NT3
         NT3=1
         CALL INVERS(1)
C  CHECK TO MAKE SURE AN ORTHOGONAL MOVE IS NOT ATTEMPTED.
         DO 110 I=1,N
         IF(DELX(I).EQ.0.0) GO TO 110
         WRITE(6,105)
     105 FORMAT(94H0 THE MATRIX OF SECOND PARTIALS IS NOT POSITIVE DEFINITE
     1, SENSITIVITY ANALYSIS IS TERMINATED    )
         GO TO 202
     110 CONTINUE
         DO 120 I=1,NPAR
         X2H(I) = PAR(I)
     120 CONTINUE
         DO 200 J=1,NPAR
         IF(NEXOPS.NE.2) GO TO 115
         IF(KTEST(J).EQ.0) GO TO 200
     115 IF(NEXOP4.EQ.0) GO TO 121
         CALL LMULT(0,DELMU,DEM,DU)
C  COMPUTE D(DEL P)/DA(J) AND D(F)/DA(J) USING CENTRAL DIFFERENCING.
     121 PAR(J) =PAR(J) + DPAR(J)
         CALL RESINT(0,DF)
         IF(M.EQ.0) GO TO 126
         DO 125 I=1,M
         CALL RESINT(I,RJ(I))
         IF(RJ(I).GT.AVAL) GO TO 125
     123 DPAR(J)=0.1*DPAR(J)
         PAR(J) = X2H(J)
         WRITE(6,124) J,DPAR(J)
     124 FORMAT(16H RESETTING DPAR(,I2,3H)= ,G14.5)
         IF(DPAR(J).EQ.0.) GO TO 201
         GO TO 121
     125 CONTINUE
     126 IF(MZ.EQ.0) GO TO 128
         DO 127 I=1,MZ
         MZPI=M+I
         CALL RESINT(MZPI,RJ(MZPI))
     127 CONTINUE
     128 IF(NEXOP4.EQ.0) GO TO 129
         CALL LMULT(1,DELMU,DEM,DU)
     129 CALL GRAD(2)
         DO 130 I=1,N
         DELX(I)=DELX0(I)
     130 CONTINUE
         DEM=2.0*DPAR(J)
         PAR(J)=PAR(J) - DEM
         CALL RESINT(0,VAL)
         IF(M.EQ.0) GO TO 136
         DO 135 I=1,M
         CALL RESINT(I,RJ(I))
         IF(RJ(I).GT.AVAL) GO TO 135
         GO TO 123
     135 CONTINUE
     136 IF(MZ.EQ.0) GO TO 138
         DO 137 I=1,MZ
         MZPI=M+I
         CALL RESINT(MZPI,RJ(MZPI))
     137 CONTINUE
```

```
138   IF(NEXCF4.EQ.0) GO TO 139
      CALL LMULT(2,DELMU,DEM,DU)
139   CALL GRAD(2)
      DF=(DF-VAL)/DEM
      DO 140 I=1,N
      DELX(I)=(DELX(I) - DELXO(I))/DEM
  140 CONTINUE
      RAF(J) = X2H(J)
C  HAVING ALREADY FACTORED A, SOLVE A*X=B FOR X,
C  WHERE B=DELX AND X=DX/DA(J).
      CALL INVERS(2)
C  PRINT OUT DX/DA(J)
      WRITE(6,150) J
  150 FORMAT(47H  X-DERIVATIVES ARE WITH RESPECT TO PARAMETER ,I2)
      DO 170 I=1,N,6
      II=MINO(I+5,N)
      WRITE(6,160) (JJ,DELX(JJ), JJ=I,II)
  160 FORMAT( 6(4H DX(,I2,2H)=,G14.7) )
  170 CONTINUE
      IF(NEXCF4.EQ.0) GO TO 375
      IF(M.EQ.0) GO TO 301
      CALL LMULT(3,DELMU,DEM,DU)
      WRITE(6,350) J
350   FORMAT(/  41H  U-DERIVATIVES WITH RESPECT TO PARAMETER    ,I2)
      DO 351 I=1,M,6
      II = MINO(I+5,M)
      WRITE(6,352) ((JJ,DU(JJ)),JJ=I,II)
352   FORMAT(6(4H DU(,I2,2H)=,G14.7))
351   CONTINUE
301   IF(MZ.EQ.0) GO TO 375
      CALL LMULT(4,DELMU,DEM,DU)
      WRITE(6,360) J
360   FORMAT(/  41H  W-DERIVATIVES WITH RESPECT TO PARAMETER    ,I2)
      DO 361 I=1,MZ,6
      II = MINO(I+5,MZ)

      WRITE(6,362) ((JJ,DU(JJ+M)), JJ=I,II)
362   FORMAT(6(4H DW(,I2,2H)=,G14.7))
361   CONTINUE
375   CONTINUE
C  COMPUTE DF/DA(J).
      DO 180 I=1,N
      DF = DF + X3H(I)*DELX(I)
  180 CONTINUE
C  PRINT DF/DA(J).
      WRITE(6,190) DF
  190 FORMAT(/ 10X,13HDF(X(R))/DA= ,G14.6/10H **********/)
  200 CONTINUE
      GO TO 202
  201 WRITE(6,106) J
  106 FORMAT(1H ,21HTERMINATING PARAMETER,I3,16H DUE TO DPAR = 0   /)
      GO TO 200
  202 NT3=NP
      CALL REJECT
      DO 205 I=1,N
      DELX(I) = DELTX(I)
  205 DELXO(I) = DELT(I)
      RETURN
      END
```

```
      SUBROUTINE LMULT(IND,DELMU,DEM,DU)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 RHOIN,RATIO,EPSI,THETAO,XEP1,XEP2
      COMMON/SHAPE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/VALUE/F,G,P0,FSIGMA,RJ(40),RHO
      COMMON/EQAL/ H, H1, MZ
      COMMON/CRST/DELX(20),DELXO(20),RHOIN,RATIO,EPSI,THETAO,
     1NTCTR,NUMINI,X1(20),X2(20),X3(20),XF2(20),XF1(20),PR1,
     2PR2,P1,F1,RJ1(40),DOTT,PGRAD(20),DIAG(20),
     3PREV3,ADELX,RSIG1,G1,NPHASE,NSATIS
      DIMENSION DELMU(40),DU(40)
      MMZ = M + MZ
      GO TO (1,2,3,3), IND
C IND = 0
      DO 100 I=1,MMZ
100   DELMU(I) = 0.
      RETURN
C IND = 1
1     DO 50 I=1,MMZ
50    DELMU(I) = RJ(I)
      RETURN
C IND = 2
2     DO 60 I=1,MMZ
60    DELMU(I) = (DELMU(I) - RJ(I))/DEM
      RETURN
3     DO 70 I=1,MMZ
      CALL GRAD1(I)
      CALL RESTNT(I,VAL)
      SUM = 0.
      DO 71 JJ=1,N
71    SUM = SUM + DEL(JJ)*DELX(JJ)
      IF(IND.EQ.4) GO TO 80
      DU(I) =-(SUM + DELMU(I))*RHO/(VAL**2)
      GO TO 70
80    DU(I) = 2.*(SUM + DELMU(I))/RHO
70    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRESEN(DU,KTEST)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/SHARE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/VALUE/F,G,FO,FSIGMA,FJ(40),RHO
      COMMON/EQAL/ H, H1,MZ
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
      DIMENSION GX(40),DU(40),KTEST(20),KLIST(20)
      MPMZ = M + MZ
      FTEST = 0.001 * DABS(F)
      DO 100 J=1,NPAR
      KTEST(J) = 0
      PAR(J) = PAR(J) + DPAR(J)
      CALL FESTNT(0,DF)
      IF(MPMZ.EQ.0) GO TO 20
      DO 10 I=1,MPMZ
      CALL FESTNT(I,DU(I))
10    CONTINUE
20    DEM = 2. * DPAR(J)
      PAR(J) = PAR(J) - DEM
      CALL FESTNT(0,XF)
      IF(MPMZ.EQ.0) GO TO 40
      DO 30 I=1,MPMZ
      CALL FESTNT(I,GX(I))
30    CONTINUE
40    DFEPS = (DF - XF)/DEM
      IF(MPMZ.EQ.0) GO TO 60
      DO 50 I=1,MPMZ
50    DU(I) = (DU(I) - GX(I))/DEM
60    SUM = DFEPS
      IF(M.EQ.0) GO TO 80
      DO 70 I=1,M
70    SUM = SUM - RHO/FJ(I)*DU(I)
80    IF(MZ.EQ.0) GO TO 95
      TSUM = 0.
      DO 90 I=1,MZ
      IM = I+M
90    TSUM = TSUM + FJ(IM)*DU(IM)
      SUM = SUM + TSUM * 2./RHO
95    DEL(J) = SUM
      PAR(J) = PAR(J) + DPAR(J)
      DTEST = DABS(DEL(J))
      IF(DTEST.GE.FTEST) KTEST(J) = 1
100   CONTINUE
      WRITE(6,600)
600   FORMAT(22X,34HOPTIMAL VALUE FUNCTION SENSITIVITY      //)
      DO 200 I=1,NPAR,5
      II=MIN0(I+4,NPAR)
      WRITE(6,601) ((JJ,DEL(JJ)), JJ=I,II)
601   FORMAT( 5(7H DF/DA(,I2,2H)=,G14.7))
200   CONTINUE
      JJ = 0
      DO 250 J=1,NPAR
      IF(KTEST(J).EQ.0) GO TO 250
      JJ = JJ + 1
      KLIST(JJ) = J
250   CONTINUE
      IF(JJ.EQ.0) GO TO 300
      WRITE(6,602)
602   FORMAT( /51H DETAILED SENSITIVITY RESULTS FOLLOW FOR PARAMETERS  )
      WRITE(6,603) (KLIST(I), I=1,JJ)
603   FORMAT(1H 40(I2,2H ,))
      WRITE(6,604)
604   FORMAT(/)
      RETURN
300   WRITE(6,605)
605   FORMAT( 42H THERE ARE NO DETAILED SENSITIVITY RESULTS      //)
      RETURN
      END
```

APPENDIX B

USER SUBROUTINES FOR SCHRADY-CHOE PROBLEM

```
      SUBROUTINE READIN
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/INV/BETA(20),PHI(20),DENSE(20),IDENT(20),NI
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
  901 FORMAT(I5,4F12.0)
      READ(5,901) NI,PAR(1),PAR(2)
      WRITE(6,901) NI,PAR(1),PAR(2)
      DO 100 I=1,NI
      READ(5,901)   IDENT(I),(PAR(4*I-2+J),J=1,4)
      WRITE(6,901)  IDENT(I),(PAR(4*I-2+J),J=1,4)
  100 CONTINUE
      NPAR = 4*NI+2
      RETURN
      END
```

```
      SUBROUTINE RESTNT(IN,VAL)
      IMPLICIT REAL*8(A-H,O-Z)

      COMMON/SHARE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/INV/BETA(20),PHI(20),DENSE(20),IDENT(20),NI
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
      VAL = 0.
      IF(IN.EQ.0) GO TO 300
      IF(IN.EQ.1) GO TO 100
  200 DO 250 I=1,NI
      IJ = 2*I-1
      QQ = X(IJ)
      IF(QQ.LE.0.) GO TO 180
  250 VAL = VAL + PAR(4*I+2)/QQ
      VAL = PAR(2) - VAL
      RETURN
  180    VAL = -1.0
      RETURN
  100 DO 150 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      RR = X(IJ1)
      IF(RR.LE.0.) GO TO 180
      QQ = X(IJ)
      IF(QQ.LT.0.) GO TO 180
  150 VAL = VAL + PAR(4*I+1)*(RR+QQ/2.-PAR(4*I-1))
      VAL = PAR(1) - VAL
      RETURN
  300 DO 350 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      QQ = X(IJ)
      RR = X(IJ1)

      UU = PAR(4*I-1)

      SS = PAR(4*I)

      DELTA = RR - UU
      ZN = DELTA / SS
      CALL ANDTR(ZN,FI,DEN)
      PHI(I) = FI
      DENSE(I) = DEN
      BETA(I) =0.5*((SS*SS+DELTA*DELTA)*PHI(I)-SS*DELTA*DENSE(I))
  350 VAL = VAL + BETA(I)/QQ
      RETURN
      END
```

```
      SUBROUTINE GRAD1(IN)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/SHARE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/INV/BETA(20),PHI(20),DENSE(20),IDENT(20),NI
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
      IF(IN.EQ.0) GO TO 300
      IF(IN.EQ.1) GO TO 100
      DO 250 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      QQ = X(IJ)
      DEL(IJ1) = 0.
  250 DEL(IJ) = PAR(4*I+2)/QQ/QQ
      RETURN
  100 DO 150 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      DEL(IJ1) = -PAR(4*I+1)
  150 DEL(IJ) = DEL(IJ1)/2.
      RETURN
  300 DO 350 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      QQ = X(IJ)
      RR = X(IJ1)

      UU = PAR(4*I-1)

      SS = PAR(4*I)

      DELTA = RR - UU
      ZN = DELTA / SS
      CALL ANDTR(ZN,FI,DEN)
      PHI(I) = FI
      DENSE(I) = DEN
      BETA(I) =0.5*((SS*SS+DELTA*DELTA)*PHI(I)-SS*DELTA*DENSE(I))
      DEL(IJ1)=(DELTA*PHI(I)-SS*DENSE(I))/QQ
  350 DEL(IJ) = -BETA(I)/QQ/QQ
      RETURN
      END

      SUBROUTINE MATRIX(IN,IKK)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/SHARE/X(20),DEL(20),A(20,20),N,M,MN,NP1,NM1
      COMMON/INV/BETA(20),PHI(20),DENSE(20),IDENT(20),NI
      COMMON/SEN/PAR(20),DPAR(20),NPAR,ISENS
      IF(IN.EQ.0) GO TO 300
      IF(IN.EQ.1) GO TO 100
      DO 250 I=1,NI
      IJ = 2*I-1
      QQ = X(IJ)
  250 A(IJ,IJ) = -2.*PAR(4*I+2)/QQ**3
      RETURN
  100 IKK = 1
      RETURN
  300 DO 350 I=1,NI
      IJ1 = 2*I
      IJ = IJ1 - 1
      QQ = X(IJ)
      RR = X(IJ1)

      UU = PAR(4*I-1)

      SS = PAR(4*I)

      DELTA = RR - UU
      ZN = DELTA / SS
      CALL ANDTR(ZN,FI,DEN)
      PHI(I) = FI
      DENSE(I) = DEN
      BETA(I) =0.5*((SS*SS+DELTA*DELTA)*PHI(I)-SS*DELTA*DENSE(I))
      A(IJ,IJ) =  2.*BETA(I)/QQ**3
      A(IJ,IJ1)=-(DELTA*PHI(I)-SS*DENSE(I))/QQ/QQ
  350 A(IJ1,IJ1) = PHI(I)/QQ
      RETURN
      END

      SUBROUTINE ANDTR(XX,PHI,DENSE)
      IMPLICIT REAL*8(A-H,O-Z)
      AX = DABS(XX)
      T = 1.D0/(1.D0+0.2316419*AX)
      DENSE = 0.3989423*DEXP(-XX*XX/2.)
      PHI = DENSE*T*((((1.330274*T -1.821256)*T+1.781478)*T
     1 - 0.3565638)*T + 0.3193815)
      IF(XX) 1,2,2
    1 PHI = 1.0 - PHI
    2 RETURN
      END
```

THE GEORGE WASHINGTON UNIVERSITY
Program in Logistics
Distribution List for Technical Papers

The George Washington University
- Office of Sponsored Research
- Library
- Vice President H. F. Bright
- Dean Harold Liebowitz
- Mr. J. Frank Doubleday

ONR
- Chief of Naval Research
  - (Codes 200, 430D, 1021P)
  - Resident Representative

OPNAV
- OP-40
- DCNO, Logistics
- Navy Dept Library
- OP-911
- OP-964

Naval Aviation Integrated Log Support

NAVCOSSACT

Naval Cmd Sys Sup Activity Tech Library

Naval Electronics Lab Library

Naval Facilities Eng Cmd Tech Library

Naval Ordnance Station
- Louisville, Ky.
- Indian Head, Md.

Naval Ordnance Sys Cmd Library

Naval Research Branch Office
- Boston
- Chicago
- New York
- Pasadena
- San Francisco

Naval Research Lab
- Tech Info Div
- Library, Code 2029 (ONRL)

Naval Ship Engng Center
- Philadelphia, Pa.
- Hyattsville, Md.

Naval Ship Res & Dev Center

Naval Sea Systems Command
- Tech Library
- Code 073

Naval Supply Systems Command
- Library
- Capt W. T. Nash

Naval War College Library
- Newport

BUPERS Tech Library

FMSO

Integrated Sea Lift Study

USN Ammo Depot Earle

USN Postgrad School Monterey
- Library
- Dr. Jack R. Borsting
- Prof C. R. Jones

US Marine Corps
- Commandant
- Deputy Chief of Staff, R&D

Marine Corps School Quantico
- Landing Force Dev Ctr
- Logistics Officer

Armed Forces Industrial College

Armed Forces Staff College

Army War College Library
- Carlisle Barracks

Army Cmd & Gen Staff College

US Army HQ
- LTC George L. Slyman
- Army Trans Mat Command

Army Logistics Mgmt Center
- Fort Lee

Commanding Officer, USALDSRA
- New Cumberland Army Depot

US Army Inventory Res Ofc
- Philadelphia

HQ, US Air Force
- AFADS-3

Griffiss Air Force Base
- Reliability Analysis Center

Maxwell Air Force Base Library

Wright-Patterson Air Force Base
- HQ, AF Log Command
- Research Sch Log

Defense Documentation Center

National Academy of Science
- Maritime Transportation Res Board Library

National Bureau of Standards
- Dr E. W. Cannon
- Dr Joan Rosenblatt

National Science Foundation

National Security Agency

WSEG

British Navy Staff

Logistics, OR Analysis Establishment
- National Defense Hdqtrs, Ottawa

American Power Jet Co
- George Chernowitz

ARCON Corp

General Dynamics, Pomona

General Research Corp
- Dr Hugh Cole
- Library

Planning Research Corp
- Los Angeles

Rand Corporation
- Library

Carnegie-Mellon University
- Dean H. A. Simon
- Prof G. Thompson

Case Western Reserve University
- Prof B. V. Dean
- Prof John R. Isbell
- Prof M. Mesarovic
- Prof S. Zacks

Cornell University
- Prof R. E. Bechhofer
- Prof R. W. Conway
- Prof J. Kiefer
- Prof Andrew Schultz, Jr.

Cowles Foundation for Research
- Library
- Prof Herbert Scarf
- Prof Martin Shubik

Florida State University
- Prof R. A. Bradley

Harvard University
- Prof K. J. Arrow
- Prof W. G. Cochran
- Prof Arthur Schleifer, Jr.

New York University
- Prof O. Morgenstern

Princeton University
- Prof A. W. Tucker
- Prof J. W. Tukey
- Prof Geoffrey S. Watson

Purdue University
Prof S. S. Gupta
Prof H. Rubin
Prof Andrew Whinston

Stanford
Prof T. W. Anderson
Prof G. B. Dantzig
Prof F. S. Hillier
Prof D. L. Iglehart
Prof Samuel Karlin
Prof G. J. Lieberman
Prof Herbert Solomon
Prof A. F. Veinott, Jr.

University of California, Berkeley
Prof R. E. Barlow
Prof D. Gale
Prof Rosedith Sitgreaves
Prof L. M. Tichvinsky

University of California, Los Angeles
Prof J. R. Jackson
Prof Jacob Marschak
Prof R. R. O'Neill
Numerical Analysis Res Librarian

University of North Carolina
Prof W. L. Smith
Prof M. R. Leadbetter

University of Pennsylvania
Prof Russell Ackoff
Prof Thomas L. Saaty

University of Texas
Prof A. Charnes

Yale University
Prof F. J. Anscombe
Prof I. R. Savage
Prof M. J. Sobel
Dept of Admin Sciences

Prof Z. W. Birnbaum
University of Washington

Prof B. H. Bissinger
The Pennsylvania State University

Prof Seth Bonder
University of Michigan

Prof G. E. P. Box
University of Wisconsin

Dr. Jerome Bracken
Institute for Defense Analyses

Prof H. Chernoff
MIT

Prof Arthur Cohen
Rutgers – The State University

Mr Wallace M. Cohen
US General Accounting Office

Prof C. Derman
Columbia University

Prof Paul S. Dwyer
Mackinaw City, Michigan

Prof Saul I. Gass
University of Maryland

Dr Donald P. Gaver
Carmel, California

Dr Murray A. Geisler
Logistics Mgmt Institute

Prof J. F. Hannan
Michigan State University

Prof H. O. Hartley
Texas A & M Foundation

Mr Gerald F. Hein
NASA, Lewis Research Center

Prof W. M. Hirsch
Courant Institute

Dr Alan J. Hoffman
IBM, Yorktown Heights

Dr Rudolf Husser
University of Bern, Switzerland

Prof J. H. K. Kao
Polytech Institute of New York

Prof W. Kruskal
University of Chicago

Prof C. E. Lemke
Rensselaer Polytech Institute

Prof Loynes
University of Sheffield, England

Prof Steven Nahmias
University of Pittsburgh

Prof D. B. Owen
Southern Methodist University

Prof E. Parzen
State University New York, Buffalo

Prof H. O. Posten
University of Connecticut

Prof R. Remage, Jr.
University of Delaware

Dr Fred Rigby
Texas Tech College

Mr David Rosenblatt
Washington, D. C.

Prof M. Rosenblatt
University of California, San Diego

Prof Alan J. Rowe
University of Southern California

Prof A. H. Rubenstein
Northwestern University

Dr M. E. Salveson
West Los Angeles

Prof Edward A. Silver
University of Waterloo, Canada

Prof R. M. Thrall
Rice University

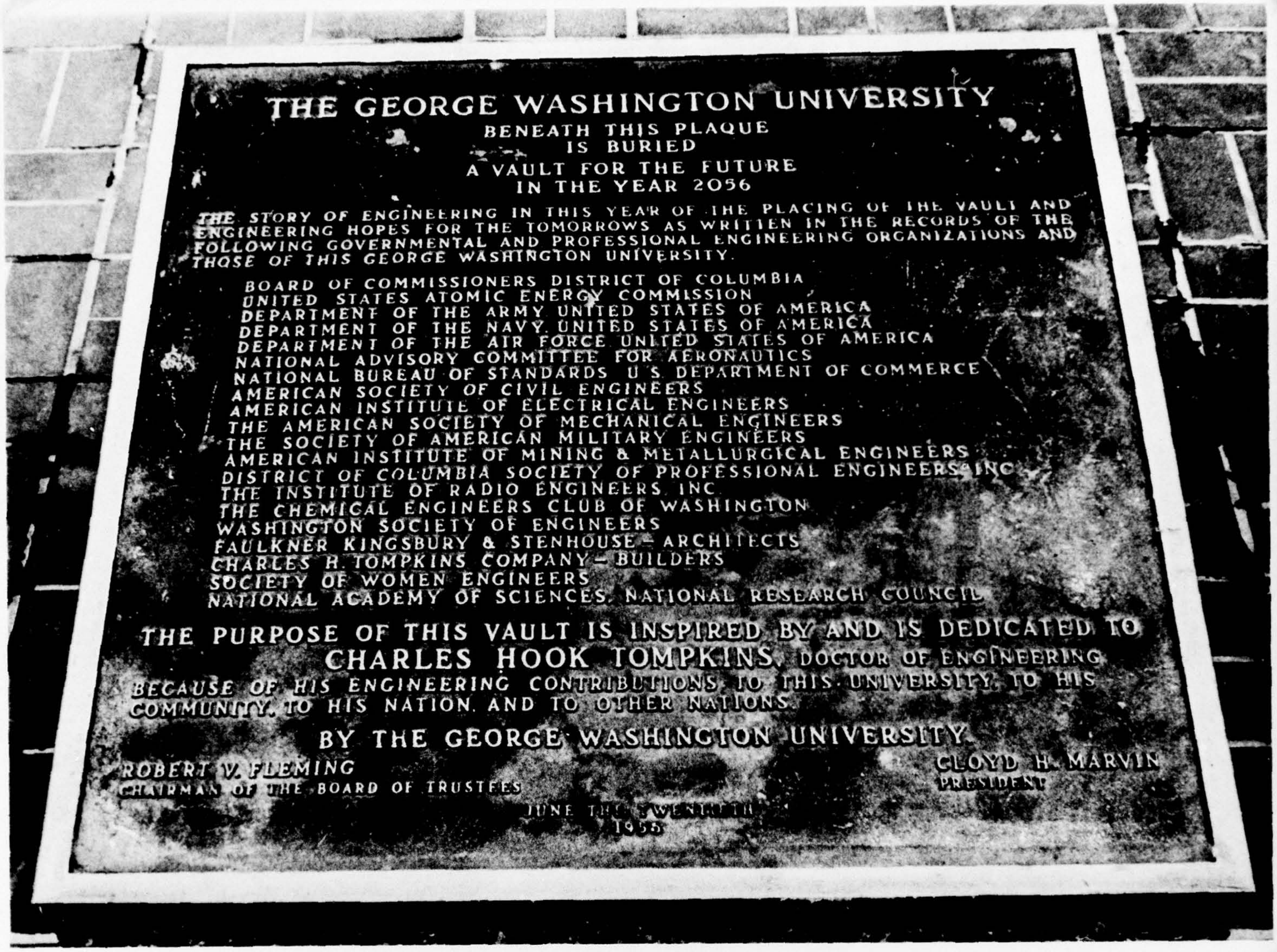
Dr S. Vajda
University of Sussex, England

Prof T. M. Whitin
Wesleyan University

Prof Jacob Wolfowitz
University of Illinois

Mr Marshall K. Wood
National Planning Association

Prof Max A. Woodbury
Duke University



To cope with the expanding technology, our society must be assured of a continuing supply of rigorously trained and educated engineers. The School of Engineering and Applied Science is completely committed to this objective.